

# Enterprise Services Bus (ESB) and Application Integration

# Ongoing Business Challenges for the State

- To reduce the cost of managing State programs while simultaneously improving the ability to manage them
- Revolutionize the delivery of services by destroying the barriers to borderless government (ie. E-Business - bridging the various interactions (G2C, G2B, G2G) with Government in a simpler way)

# Why Application/Data Integration?

- Remains one of the top business drivers for CIO's
- Companies have invested heavily in homegrown applications and business application suites (SAP, Oracle, PeopleSoft, Siebel, etc)
- Significant business benefits to linking disparate business systems
- Historically application integration solutions have been expensive, proprietary and prohibitively complex

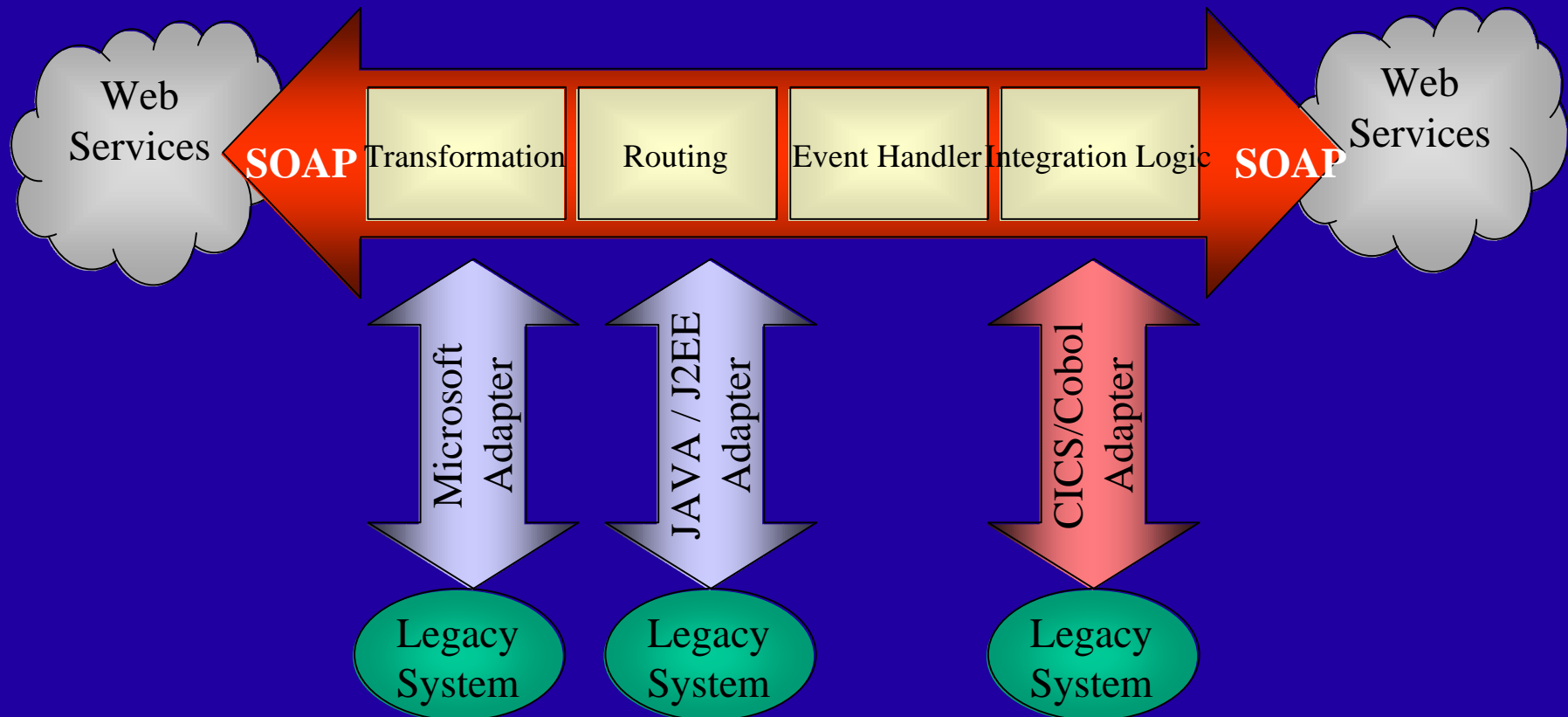
## Why An ESB??

- Uses existing technology
- Standards based
- Non-proprietary
- Uses existing messaging middleware (HTTP, Biztalk, MQ Series, JMS)
- Easy to use and deploy
- Low implementation and management costs

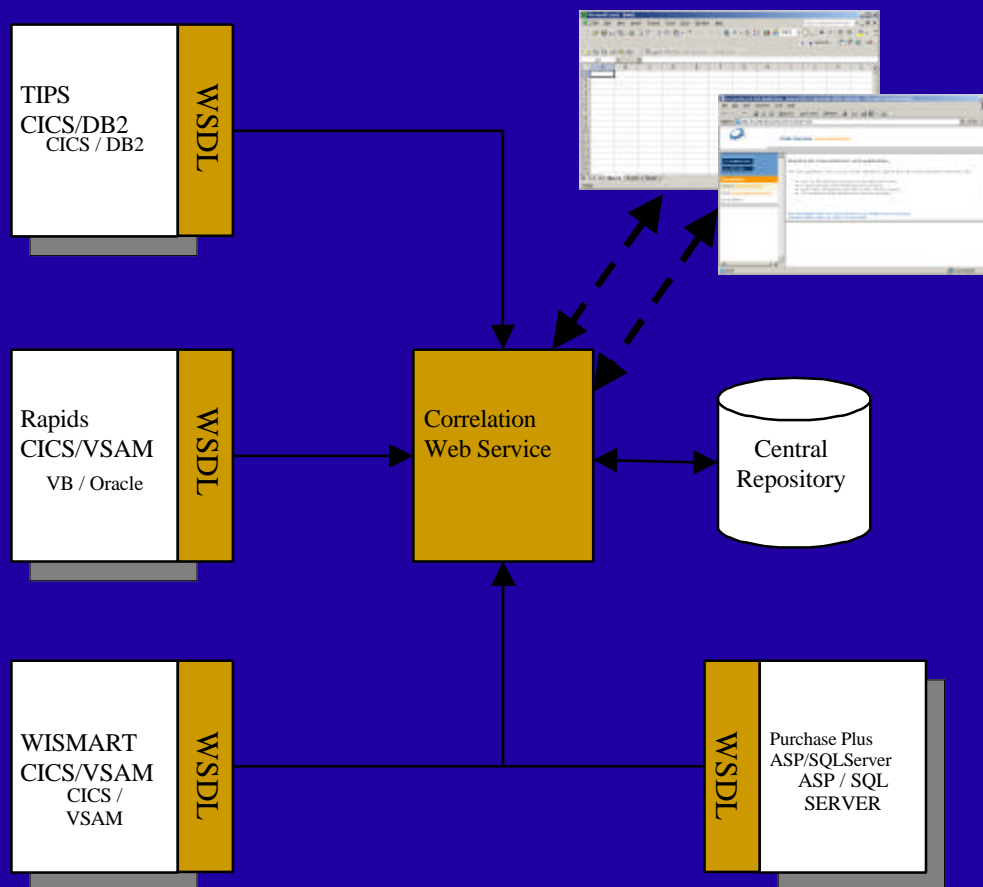
# What is an ESB?

- Standards-based Integration Backbone
  - XML, XSLT, XPATH, Java, Javascript
  - Message-Oriented Middleware (MOM)
  - Web Services
  - Transformation and Routing Intelligence
- Similar concept to application integration, but the underlying architecture is very different
  - Less Expensive
  - Non-Proprietary
  - Simpler Implementation

# ESB Overview



# DOA ESB Pilot

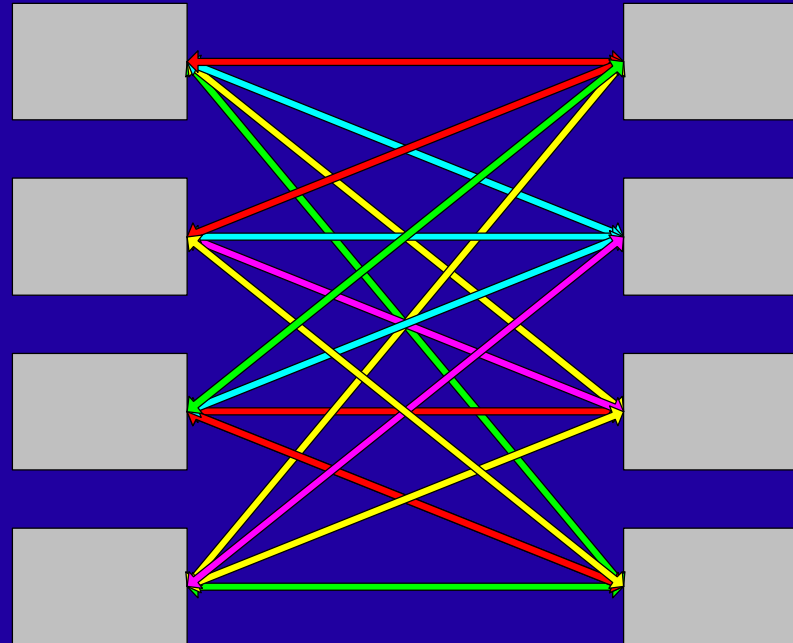


# The Potential of Enterprise Service Bus Integration



# Integration Today

- System-driven integration results in system-specific solutions using wide range of technologies
- A solution is built for each system



# Consequences

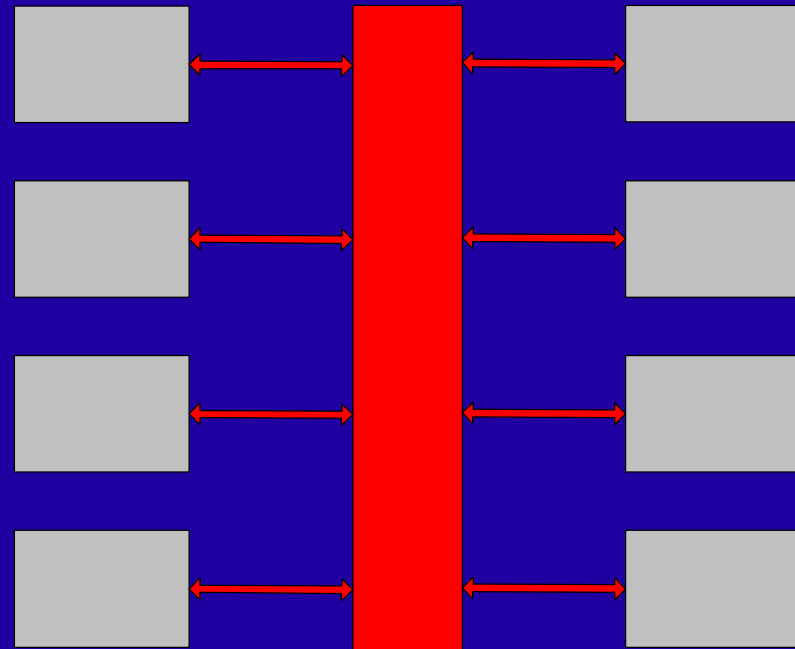
- Redundancy - Many solutions to a common problem
- Complexity - Recurrent, changing demands on tech resources
- Fragility - Volatile, strong coupling of apps and data
- Inefficiency - Repeated investment in integration infrastructure
- Low ROI Potential- Poorly suited to reuse
- Disabling - Cost & complexity inhibits sharing resources
- Unmanageable - No enterprise perspective

## Examples

- [www.wisconsin.gov](http://www.wisconsin.gov) - Integration of agency data via batch conversion processes across multiple databases
- DNR / RVR - Real-time web system w/ batch FTP transmission of transaction data between agencies
- EMSS - HTTP “page scraping” for background checks
- DRL License Renewal - Remote stored procedure invocation across firewalls via JDBC
- Local Government Sites - land record data integration effort abandoned

# Integration with ESB

- Enterprise-driven integration results in an enterprise solution using a standard set of technologies
- A solution is built for all systems to share



# Consequences

- Consistency - A single solution to a common problem
- Stability - Predictable, uniform demands on tech resources
- Durability - Flexible, weak coupling of apps and data
- Efficiency - A single investment in integration infrastructure
- Excellent ROI Potential - Exceptionally suited to reuse
- Enabling - Encourages sharing resources
- Manageable - Clear enterprise perspective

# Potential

- Procurement - Enterprise data integration
- Licensing Applications - Collaborative, distributed application development and deployment
- E-Receipts - Enterprise service sharing
- Tax Calculation - Enterprise enablement of common business processes
- Criminal Background Checks - Enterprise service sharing
- Integration across government and public sector boundaries

# Questions